IAES International Journal of Artificial Intelligence (IJ-AI)
Vol. 8, No. 3, September 2019, pp. 228~236
ISSN: 2252-8938, DOI: 10.11591/ijai.v8.i3.pp228-236

Performance comparison of various probability gate assisted binary lightning search algorithm

Md Mainul Islam[1], Hussain Shareef [2], Mahmood Nagrial[3], Jamal Rizk[4] and Ali Hellany[5], Saiful Nizam Khalid[6]
[1,3,4,5]School of Computing, Engineering and Mathematics, Western Sydney University, Locked Bag 1797, NSW 2751, Australia
[2]Department of Electrical Engineering, United Arab Emirates University, 15551, 1 Al-Ain, United Arab Emirates
[6]Department of Electrical Power Engineering, Universiti Teknologi Malaysia, 81310 Johor Bahru, Malaysia

Article Info

Article history:

Received May 5, 2019
Revised Jun 19, 2019
Accepted Jul 27, 2019

Keywords:

Benchmark function
Binary lightning search algorithm
Decoding
Probability
Transfer functions

ABSTRACT

Recently, many new nature-inspired optimization algorithms have been introduced to further enhance the computational intelligence optimization algorithms. Among them, lightning search algorithm (LSA) is a recent heuristic optimization method for resolving continuous problems. It mimics the natural phenomenon of lightning to find out the global optimal solution around the search space. In this paper, a suitable technique to formulate binary version of lightning search algorithm (BLSA) is presented. Three common probability transfer functions, namely, logistic sigmoid, tangent hyperbolic sigmoid and quantum bit rotating gate are investigated to be utilized in the original LSA. The performances of three transfer functions based BLSA is evaluated using various standard functions with different features and the results are compared with other four famous heuristic optimization techniques. The comparative study clearly reveals that tangent hyperbolic transfer function is the most suitable function that can be utilized in the binary version of LSA.

Corresponding Author:

Hussain Shareef
Department of Electrical Engineering,
United Arab Emirates University,
15551, 1 Al-Ain, United Arab Emirates.
Email: shareef@uaeu.ac.ae

1. INTRODUCTION

In real-world condition, most of the problems are highly nonlinear. Over the last few decades, nature-based optimization algorithms have become powerful optimization tools for solving high dimensional nonlinear real-world problems in the area of science, commerce and engineering [1-3]. As no optimization algorithms can guarantee best result for all kinds of practical problems [4] the mimicking sense from nature always inspire to develop new optimization technique. One of the most recent nature-inspired optimization algorithms is the LSA first presented by Shareef et al. [5] based on natural occurrence of lightning. The basic LSA chiefly works in continuous search environment to resolve real area constrained optimization difficulties. Nevertheless, numerous optimization problems are vital to set in binary manner and solve them in binary search space by choosing “0” or “1” as well as seldom the outcomes need to be exhibited with some bits. Therefore, binary optimization algorithms are needed to solve those problems. In practical, the binary optimization algorithms are successfully utilized in many applications such as feature selection and modeling [6], unit commitment problems [7], power quality monitor placement [8], micro-grid operation scheduling [9], power quality and reliability enhancement [10], flow shop scheduling [11], optimum power flow problem [12], knapsack problem [13], optimal placement of sensors [14] and so on. The typical continuous domain optimization algorithm can be converted to binary one. The key factor between the continuous and binary domain optimization algorithms is the transfer function. This function maps the binary search space from continuous one depending on the function's probability condition. The most common transfer functions, namely, log sigmoid, tangent hyperbolic sigmoid and quantum bit rotating gate are generally applied for updating process of the continuous version of algorithms. The log sigmoid activation function has been successfully applied in many binary domain optimization algorithms such as binary particle swarm optimization (BPSO) [15], binary firefly (BFA) [11], binary cuckoo search [16] and binary bacterial foraging [17]. Similarly, tangent hyperbolic sigmoid activation function is utilized to form binary gravitational search algorithm (BGSA) [18], binary artificial bee colony [19] and binary magnetic optimization [20]. Nonetheless, quantum bit is promoted on existing binary algorithm for performance improvement such as quantum-inspired binary particle swarm optimization [7], quantum-inspired binary gravitational search algorithm [21], quantum-inspired binary firefly algorithm [10], quantum-inspired tabu search algorithm [22] and etc. The aim of this study is to formulate a binary version of LSA and identify which transfer function is most suitable to modify traditional LSA to handle binary optimization problems. Finally, the performances of those transfer functions are distinguished in terms of correctness of results and conjunction speed.

2. BINARY LIGHTNING ALGORITHM (BLSA)

The binary lightning search algorithm (BLSA) is first introduced by Islam et al. [23] to elevate the binary combinational difficulties relaying on the binary search space. In BLSA, three projectile forms are used: the transition projectiles that generate the first step leader population, the space projectiles (p_i^S) that try to become the leader, and the lead projectile (p^L) that signifies the projectile excited from the best-positioned step leader. The updating process of space projectile is presented as

$$p_{i\_new}^S = p_i^S \pm exprand(\mu_i) \tag{1}$$

where *exprand* is an exponential random integer, and μ_i signifies the distance among the lead projectile p^L and the space projectile p_i^S under study.

The new position of p^L at *step+1* can be expressed as

$$p_{i\_new}^L = p_i^L + normrand(\mu_{L,}\sigma_L) \tag{2}$$

where *normrand* is a random number caused by the normal distribution function, μ_L is occupied as current value of p^L, and σ_L is the scale parameter that exponentially declines as it discover the best solution. The full BLSA can be found in [23]. The following subsections, explains their adaptation in making the binary version of LSA.

2.1. Logistic sigmoid assisted updating

In this paper the probability transfers function $T_f(p_i)$, taken as a probability gate, is calculated based on (3) to plan a binary search space. Conversely, the projectile's position at *step+1* is upgraded succeeding the probability function with a terms as shown in (4). This formulation must exhibit less possibility to alter the projectile position if $|p_i|$ is small. It also should be capable to make great possibility of altering the projectile location with bigger value of $|p_i|$. Logistic sigmoid transfer function has this feature as shown in (3).

$$T_f(p_i) = \frac{1}{1+e^{-p_i}} \tag{3}$$

$$P_{i\_new} = \begin{cases} \overline{p_i}, & if \ rand \le |T_f(p_i)| \\ p_i, & otherwise \end{cases} \tag{4}$$

where *rand* is the uniform arbitrary variable in interval [0, 1].

2.2. Tangent hyperbolic sigmoid assisted updating

Like the logistic sigmoid probability transfer function, Tangent hyperbolic activation function can be utilized in the updating process as shown in (5) to drawn to a binary search universe. Conversely, the projectile's position at *step+1* is upgraded considering the probability function with a terms as shown in (6). In (5) suggests that when the $|p_i|$ are large from a specific position, the probability of directing p_i in (6) to a new location at *step+1* is high, whereas by decreasing the $|p_i|$ at later iterations, the probability of directing p_i is reduced and lastly when the $|p_i|$ is zero, the position of p_i remains unaffected.

$$T_f(p_i) = |\tanh\ (p_i)| \tag{5}$$

$$P_{i\_new} = \begin{cases} \overline{p_i}, & if\ rand \le |T_f\ (p_i)| \\ p_i, & otherwise \end{cases} \tag{6}$$

where *rand* is the uniform random variable in interval [0, 1].

2.3. Quantum bit assisted updating

A quantum bit (Q-bit) is the lowest unit in quantum computing in which it stores the data in dual-state. This Q-bit can be in the "0" state, in the "1" state or in a linear superposition of two [24]. The state of Q-bit can be represented as follows:

$$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle \tag{7}$$

where α and β are complex numbers that fulfill the probability magnitudes of the resulting states. Hence, the states can be normalized to 1 as shown [24]:

$$|\alpha|^2 + |\beta|^2 = 1 \tag{8}$$

The Q-bit state is upgraded by using a quantum gate, known as alterable gate, and can be characterized as a unitary operator *U*.

$$U(\Delta\theta_i) = \begin{bmatrix} \cos(\Delta\theta_i) & -\sin(\Delta\theta_i) \\ \sin(\Delta\theta_i) & \cos(\Delta\theta_i) \end{bmatrix} \tag{9}$$

where $\Delta\theta_i$, i= 1,2,3,…,n denotes the rotation angle of each Q-bit toward either the "0" or "1" state depending on the Q-bit sign. The rotation angle can be expressed as

$$\Delta\theta_i = \theta \times p_i \tag{10}$$

where θ is the rotation angle amplitude beside the iteration and decline monotonously from $\theta_{\max}$ to $\theta_{\min}$ as shown in (11). The Q-bit individual sequence is then upgraded based on the rotation angle and rotation gate (12). Finally, the projectile's position is upgraded based on the probability of $|\beta|^2$ as shown in (13).

$$\theta = \theta_{max} - (\theta_{max} - \theta_{min}) \times \frac{step_{current}}{step_{max}} \tag{11}$$

$$\begin{bmatrix} \alpha_i(step+1) \\ \beta_i(step+1) \end{bmatrix} = U(\Delta\theta_i) \times \begin{bmatrix} \alpha_i(step) \\ \beta_i(step) \end{bmatrix} \tag{12}$$

$$P_{i\_new} = \begin{cases} 1, & if\ rand < |\beta_i\ (step+1)|^2 \\ 0, & otherwise \end{cases} \tag{13}$$

where *rand* is the even arbitrary variable in interval [0, 1].
The following are the steps of BLSA algorithm.
Step 1: The algorithm is initialized by declaring the iteration, channel period and leader tips energies.

Step 2: Arbitrarily create step leaders (transition projectile).
Step 3: Evaluate the performance (projectiles energies).
Step 4: Update leader tip energies as well as the best and worst step leaders.
Step 5: Check maximum channel time. If maximum channel time occurs, replace the worst step leader with the best one.
Step 6: Then, update each projectile position using (4) or 6 or 13 and evaluate the performance.
Step 7: Check focking occurs or not after that update step leaders.
Step 8: Repeat Steps 4 to 7 until the optimization criteria is achieved. In this study, convergence criterion is based on maximum iteration number.
Step 9: Print solution.

3. PERFORMANCE EVALUATION USING BENCHMARK FUNCTIONS

3.1. Benchmark function

Benchmark function plays an important role for testing and validating new optimization algorithm. It governs the competence and effectiveness of the algorithm. To measure the effectiveness of BLSA, a range of benchmark functions are efficiently applied. It contains different search spaces, dimensions and global optimal are shown in Tables 1 to 3. The benchmark functions are classified as follows:

3.1.1. Unimodal functions

The main characteristics of benchmark functions F1 and F2 used in this study are high dimensional functions and contain only one global optimal for each function. Table 1 represents the properties of utilized unimodal functions in detail.

Table 1. Unimodal test functions

ID	Test function	Name of Function	Search range	Dimension(n)	Best known
F1	$f(\mathbf{x}) = \sum_{i=1}^{n} x_i^2$	Sphere	$[-100,100]^n$	30	0
F2	$f(\mathbf{x}) = \sum_{i=1}^{n} \lvert x_i \rvert + \prod_{i=1}^{n} \lvert x_i \rvert$	Schwefel 2.22	$[-10,10]^n$	30	0

3.1.2. High dimensional multimodal functions

The high dimensional functions F3 and F4 are considered as the most challenging problems because it contains many local minima. Table 2 presents the properties of applied high dimensional multimodal functions.

Table 2. High dimensional multimodal test functions

ID	Test function	Name of Function	Search range	Dimension	Best known
F3	$f(\mathbf{x}) = \sum_{i=1}^{n} \left[x_i^2 - 10\cos(2\pi x_i) + 10 \right]$	Rastrigin	$[-5.12,5.12]^n$	30	0
F4	$f(\mathbf{x}) = -20\exp\left(-0.2\sqrt{\frac{1}{n}\sum_{i=1}^{n} x_i^2} \right) - \exp\left(\frac{1}{n}\sum_{i=1}^{n} \cos(2\pi x_i) \right) + 20 + e$	Ackley	$[-32,32]^n$	30	0

3.1.3. Binary functions

This function group is basically consisted of binary domain test function where global optimum depends on the function dimension. Table 3 reveals the properties of utilized binary test functions.

Table 3. Binary test functions

ID	Test function	Name of Function	Search range	Dimension (n)	Best known
F5	$f(x)=\sum_{i=1}^{n} x_i$	Max one	$[0,1]^n$	80	80
F6	$f(x)=\sum_{i=1}^{\frac{n}{8}} (\prod_{j=8(i-1)+1}^{8i} x_j)$	Royal road	$[0,1]^n$	80	10

4. PERFORMANCE TESTS

In this section, the performance of proposed BLSA with various bit updating procedures is shown using different benchmark functions and compared with other four heuristic optimization techniques to validate the results.

4.1. Parameter settings

Like other heuristic optimization techniques, BLSA has its own algorithm dependent parameter for its proper functioning. The performance of BLSA is generally influenced by the energy Ept. In this study, the value of Ept is set as 1.05. To validate the results, the effectiveness of BLSA is validated with BPSO [15], BFA [11], genetic algorithm (GA) [25] and BGSA [18]. To make fair comparison, the population size and maximum iteration are set as 50 and 500, respectively, for all algorithms.

4.2. Test results and discussion

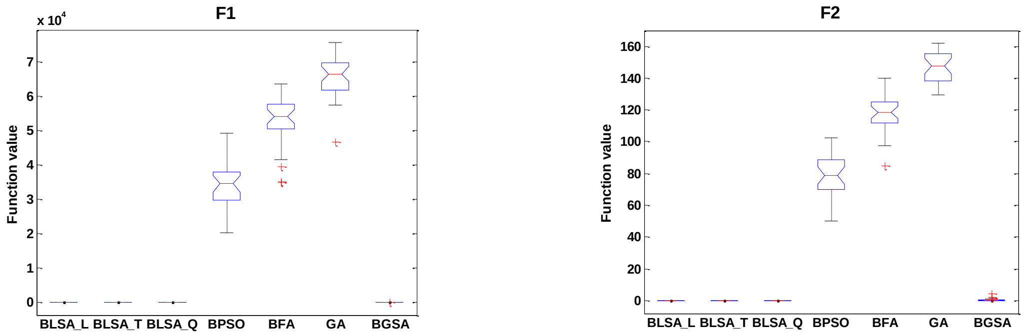
The simulations are conducted to solve the above-mentioned benchmark functions under various test function groups in order to evaluate the performance of BLSA with each transfer function in which BLSAlog (BLSA_L), BLSAtan (BLSA_T) and BLSAQbit (BLSA_Q) represent the log sigmoid, tan hyperbolic sigmoid and Quantum bit, respectively, based BLSA. Among three test cases, Tests 1 and 2 are related to objective function minimization whereas Test 3 belongs to objective function maximization.

4.2.1. Test 1

This test group is consisted of unimodal benchmark functions F1 and F2 which are high dimensional in nature. The functions F1 and F2 are comparatively easy to solve because of single global optimal. Each benchmark function is executed 30 times independently for all algorithms while the dimension of these test functions is set as 30. A comparative study is shown in Table 4 in terms of best, worst, average, median, standard deviation and minimum number of iteration. The best performance for each algorithm is weighted in bold face. From Table 4, it is shown that the BLSA can find the best known value with all bit updating mechanisms. However, BPSO, BFA, GA and BGSA cannot find the global optimal solution. The results of these algorithms are further illustrated in box plots as shown in Figure 1 relied on 30 independent runs and the convergence characteristics are depicted in Figure 2. From Table 4 and Figure 2, it is clearly observed that the tangent hyperbolic sigmoid activation function based BLSA (BLSAtan) required less number of iterations to reach the global optimal solution. In some cases, BPSO as shown in Table 4 requires less number of iterations, however, it cannot find the best known value.

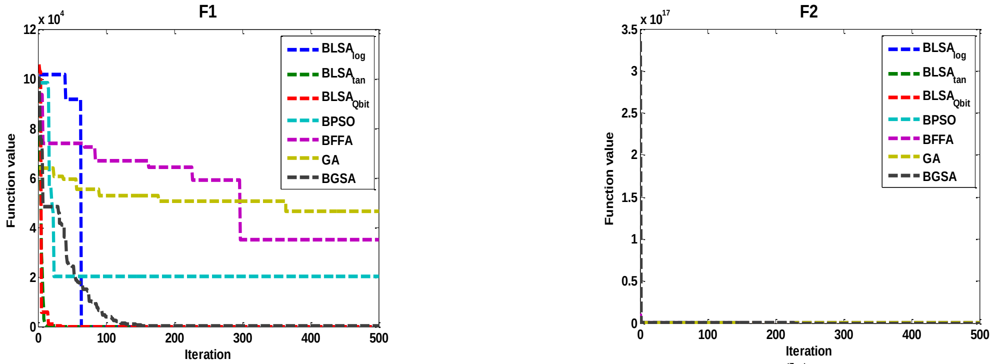
Table 4. Global optimization results of Test 1 for benchmark function in Table 1

Test Fn.	Algorithm	Best	Worst	Average	Median	Standard Deviation	Minimum iteration
F1	$BLSA_{log}$	0	0	0	0	0	62
	$BLSA_{tan}$	0	0	0	0	0	28
	$BLSA_{Qbit}$	0	0	0	0	0	34
	BPSO	20239.97	49226.20	33774.02	34494.64	7030.68	23
	BFA	34966.2	63715.34	52878.02	54068.65	7665.34	297
	GA	46584.69	75625.58	66088.58	66521.42	6120.112	364
	BGSA	1.53E-05	17.03518	1.196955	0.012367	3.292828	373
F2	$BLSA_{log}$	0	0	0	0	0	94
	$BLSA_{tan}$	0	0	0	0	0	20
	$BLSA_{Qbit}$	0	0	0	0	0	36
	BPSO	50.07819	102.6368	79.53768	78.85152	13.00522	21
	BFA	84.82404	139.8556	117.5649	118.5391	11.91653	371
	GA	129.2736	161.879	146.3888	147.5646	10.14098	56
	BGSA	0.011719	4.082031	0.598079	0.259764	0.874547	385



(a) (b)

Figure 1. Global optimization results of Test 1 for benchmark functions F1 to F3. (a) F1 and (b) F2



(a) (b)

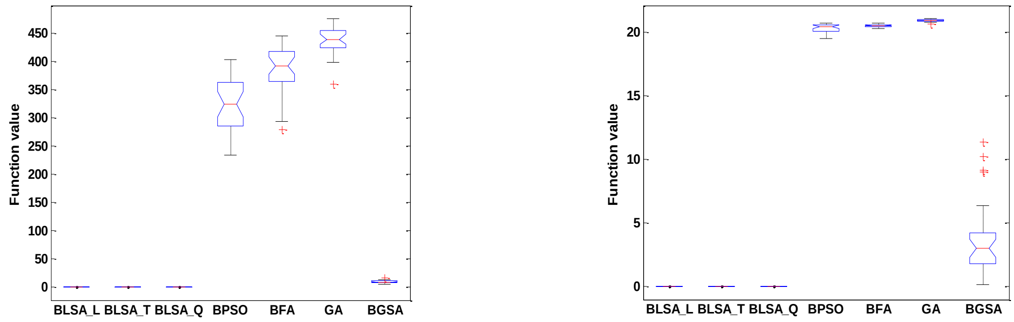
Figure 2. Convergence characteristic curves of Test 1. (a) F1 and (b) F2

4.2.2. Test 2

In this test, exploration and exploitation abilities of the BLSA are tested using multimodal high dimensional functions. It is important to note that the complexity level of the problems have increased because of containing many local minima. Due to several local minima, it is usually hard to find the best optimal solution. Therefore, final optimal outcomes of the tested algorithm are more significant. Similar to Test 1, again each benchmark function is performed 30 times while function dimension is maintained as 30. The numerical results are tabulated in Table 5. The best performance for each algorithm is weighted in boldface. The outcomes of 30 independent runs are depicted in Figure 3 through box plots. Furthermore, the convergence characteristic curves are shown in Figure 4. Similar to previous test, the tangent hyperbolic sigmoid activation function updating procedure (BLSAtan) is faster to find the global optimal solution than the others.

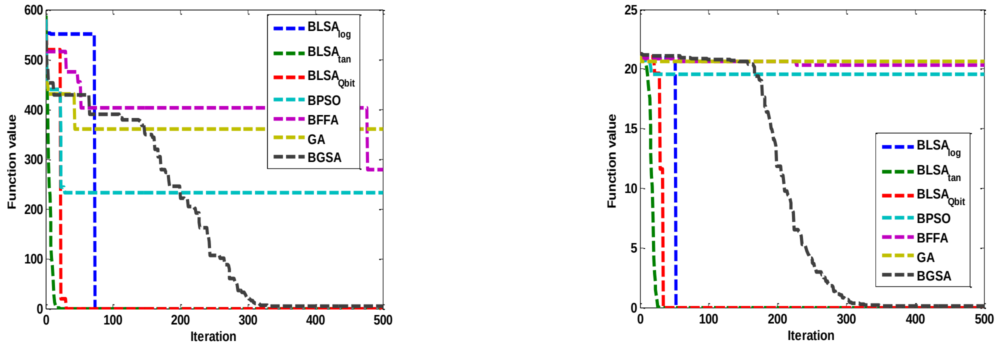
Table 5. Global optimization results of Test 2 for benchmark function in Table 2

Test Fn.	Algorithm	Best	Worst	Average	Median	Standard Deviation	Minimum iteration
	$BLSA_{log}$	0	0	0	0	0	74
	$BLSA_{tan}$	0	0	0	0	0	21
	$BLSA_{Qbit}$	0	0	0	0	0	30
F3	BPSO	232.9097	403.0246	320.4552	323.8423	42.55152	28
	BFA	278.638	444.8202	387.1754	392.0954	39.73813	477
	GA	359.775	474.774	436.5995	438.7085	25.06685	44
	BGSA	3.811253	15.94872	8.651355	8.373049	2.764086	384
	$BLSA_{log}$	8.88E-16	8.88E-16	8.88E-16	8.88E-16	1E-31	52
	$BLSA_{tan}$	8.88E-16	8.88E-16	8.88E-16	8.88E-16	1E-31	28
	$BLSA_{Qbit}$	8.88E-16	8.88E-16	8.88E-16	8.88E-16	1E-31	34
F4	BPSO	19.56376	20.72198	20.34145	20.44813	0.310445	17
	BFA	20.34612	20.72151	20.54059	20.55727	0.105571	227
	GA	20.6778	21.08882	20.96458	20.96271	0.08477	2
	BGSA	0.144861	11.39533	3.744453	2.980169	2.872996	385



(a) (b)

Figure 3. Global optimization results of Test 2 for benchmark functions F3 and F4. (a) F3 and (b) F4



(a) (a)

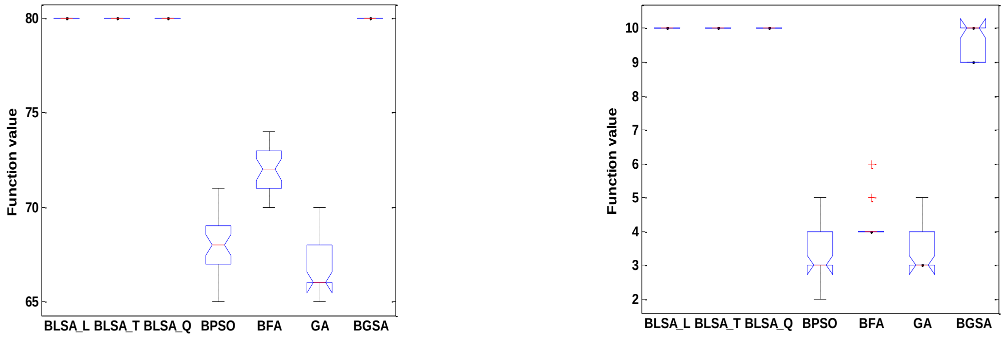
Figure 4. Convergence characteristic curves of Test 2. (a) F3 and (b) F4

4.2.3. Test 3

To determine the capability of the BLSA for maximization problem, this test group is handled with binary nature (Max one and Royal road) test function. The dimensions of these functions are set as 80. The simulation results are summarized in Table 6 based on 30 independent runs. The best performance for each algorithm is weighted in boldface. Similar to Tests 1 and 2, the performance of BLSA are identical for various bit updating procedures with different transfer functions as shown in Table 6 and Figures 5 and 6. For maximization problems, convergence characteristic curves portrayed in Figure 6 clearly represents that the quantum bit assisted BLSA (BLSAQbit) outperforms the others.

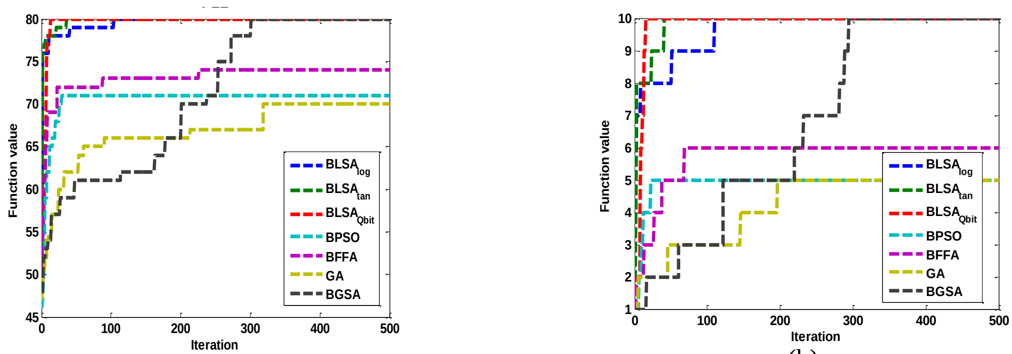
Table 6. Global optimization results of Test 3 for benchmark function in Table 3

Test Fn.	Algorithm	Best	Worst	Average	Median	Standard Deviation	Minimum iteration
F5 n=80	$BLSA_{log}$	80	80	80	80	0	104
	$BLSA_{tan}$	80	80	80	80	0	36
	$BLSA_{Qbit}$	80	80	80	80	0	13
	BPSO	71	65	67.6	68	1.499425	30
	BFA	74	70	71.93333	72	1.048261	227
	GA	70	65	66.73333	66	1.201532	320
	BGSA	80	80	80	80	0	302
F6 n=80	$BLSA_{log}$	10	10	10	10	0	110
	$BLSA_{tan}$	10	10	10	10	0	41
	$BLSA_{Qbit}$	10	10	10	10	0	15
	BPSO	5	2	3.466667	3	0.62881	23
	BFA	6	4	4.233333	4	0.504007	69
	GA	5	3	3.5	3	0.572351	197
	BGSA	10	9	9.566667	10	0.504007	295



(a) (a)

Figure 5. Global optimization results of Test 3 for benchmark functions F5 and F6. (a) F5 and (b) F6



(a) (b)

Figure 6. Convergence characteristic curves of Test 3. (a) F5 and (b) F6

5. CONCLUSION

This paper investigated different methods to realize binary version of LSA using logistic, tangent hyperbolic and quantum bit probability gate, respectively. These activation functions modified the position update procedure of the original version of LSA. The solution finding ability and consistency of various transfer functions assisted BLSA were tested with various benchmark functions of different complexity. The performance of BLSA is compared with other heuristic optimization techniques. The comparative study reveals that the performance of BLSA with tangent hyperbolic sigmoid function projectile and step leader updating procedure is the best to be utilized for minimization problems whereas quantum bit method is better for maximization problems. Moreover, it can also be concluded that both tangent hyperbolic and quantum bit assisted BLSA can be equally used for both minimization and maximization problems since they can provide best global optimal solution for all standard benchmark functions corresponding to top known value without significant differences in convergence speed and computational burdens.

REFERENCES

[1] X. Huang, X. Zeng, R. Han, and X. Wang, "An enhanced hybridized artificial bee colony algorithm for optimization problems," *IAES International Journal of Artificial Intelligence (IJ-AI)*, vol. 8, no. 1, p. 8, 2019.

[2] A. Ullah, N. M. Nawi, J. Uddin, S. Baseer, and A. H. Rashed, "Artificial bee colony algorithm used for load balancing in cloud computing: review," *IAES International Journal of Artificial Intelligence (IJ-AI)*, vol. 8, no. 2, p. 12, 2019.

[3] R. M. Amar Hamzi, "Cat Swarm Optimization to Shunt Capacitor Allocation inAlgerian Radial Distribution Power System," *IAES International Journal of Artificial Intelligence (IJ-AI)*, vol. 7, no. 3, p. 10, 2018.

[4] D. H. Wolpert and W. G. Macready, "No free lunch theorems for optimization," *IEEE Transactions on Evolutionary Computation*, vol. 1, no. 1, pp. 67-82, 1997.

[5] H. Shareef, A. A. Ibrahim, and A. H. Mutlag, "Lightning search algorithm," *Applied Soft Computing*, vol. 36, pp. 315-333, 2015/11/01/ 2015.

[6] Q. Shen, J. H. Jiang, C. X. Jiao, G. L. Shen, and R. Q. Yu, "Modified particle swarm optimization algorithm for variable selection in MLR and PLS modeling: QSAR studies of antagonism of angiotensin II antagonists," (in eng),

European journal of pharmaceutical sciences : official journal of the European Federation for Pharmaceutical Sciences, vol. 22, no. 2-3, pp. 145-52, Jun 2004.
[7] Y. Jeong, J. Park, S. Jang, and K. Y. Lee, "A New Quantum-Inspired Binary PSO: Application to Unit Commitment Problems for Power Systems," *IEEE Transactions on Power Systems,* vol. 25, no. 3, pp. 1486-1495, 2010.
[8] A. M. Ahmad Asrul Ibrahim, Hussain Shareef, "A Novel Quantum-inspired Binary Gravitational Search Algorithm in Obtaining Optimal Power Quality Monitor Placement.," *Journal of Applied Sciences,* vol. 12, pp. 822-830, 2012.
[9] P. Li, Z. Zhou, X. Lin, X. Yang, and X. Niu, "Dynamic optimal operation scheduling of microgrid using binary gravitational search algorithm," in *2014 International Conference on Power System Technology*, 2014, pp. 3175-3180.
[10] H. Shareef, A. A. Ibrahim, N. Salman, A. Mohamed, and W. Ling Ai, "Power quality and reliability enhancement in distribution systems via optimum network reconfiguration by using quantum firefly algorithm," *International Journal of Electrical Power & Energy Systems,* vol. 58, pp. 160-169, 2014/06/01/ 2014.
[11] R. R. Mohammad Kazem Sayadi, Nader Ghaffari-Nasab, "A discrete firefly meta-heuristic with local search for makespan minimization in permutation flow shop scheduling problems," *International Journal of Industrial Engineering Computations,* vol. 1, no. 1, pp. 1-10, 2010.
[12] V. Krishnasamy, "Genetic algorithm for solving optimal power flow problem with UPFC," *International Journal of Software Engineering and Its Applications,* vol. 5, no. 1, pp. 39-50, 2011.
[13] A. Gherboudj, A. Layeb, and S. Chikhi, "Solving 0-1 knapsack problems by a discrete binary version of cuckoo search algorithm," *Int. J. Bio-Inspired Comput.,* vol. 4, no. 4, pp. 229-236, 2012.
[14] G. Feng, M. Liu, and G. Wang, "Genetic algorithm based optimal placement of PIR sensors for human motion localization," *Optimization and Engineering,* journal article vol. 15, no. 3, pp. 643-656, September 01 2014.
[15] J. Kennedy and R. C. Eberhart, "A discrete binary version of the particle swarm algorithm," in *1997 IEEE International Conference on Systems, Man, and Cybernetics. Computational Cybernetics and Simulation*, 1997, vol. 5, pp. 4104-4108 vol.5.
[16] D. Rodrigues *et al.*, "BCS: A Binary Cuckoo Search algorithm for feature selection," in *2013 IEEE International Symposium on Circuits and Systems (ISCAS2013)*, 2013, pp. 465-468.
[17] B. Niu and Y. Bi, "Binary bacterial foraging optimization for 0/1 knapsack problem," in *2014 IEEE Congress on Evolutionary Computation (CEC)*, 2014, pp. 647-652.
[18] E. Rashedi, H. Nezamabadi-pour, and S. Saryazdi, "BGSA: binary gravitational search algorithm," *Natural Computing,* journal article vol. 9, no. 3, pp. 727-745, September 01 2010.
[19] C. P. G. Manasarani Mandala, "Binary Artificial Bee Colony Optimization for GENCOs'Profit Maximization under Pool Electricity Market " *International Journal of Computer Applications,* vol. 90, no. 19, pp. 34-42, 2014.
[20] S. Mirjalili and S. Z. M. Hashim, "BMOA: binary magnetic optimization algorithm," *International Journal of Machine Learning and Computing,* vol. 2, no. 3, p. 204, 2012.
[21] A. A. Ibrahim, A. Mohamed, and H. Shareef, "Optimal power quality monitor placement in power systems using an adaptive quantum-inspired binary gravitational search algorithm," *International Journal of Electrical Power & Energy Systems,* vol. 57, pp. 404-413, 2014/05/01/ 2014.
[22] Y.-H. Chou, C.-H. Chiu, and Y.-J. Yang, "Quantum-inspired tabu search algorithm for solving 0/1 knapsack problems," presented at the Proceedings of the 13th annual conference companion on Genetic and evolutionary computation, Dublin, Ireland, 2011.
[23] M. M. Islam, H. Shareef, A. Mohamed, and A. Wahyudie, "A binary variant of lightning search algorithm: BLSA," *Soft Computing,* journal article vol. 21, no. 11, pp. 2971-2990, June 01 2017.
[24] K.-H. Han and J.-H. Kim, "Quantum-inspired evolutionary algorithm for a class of combinatorial optimization," *Trans. Evol. Comp,* vol. 6, no. 6, pp. 580-593, 2002.
[25] R. L. H. a. S. E. Haupt, "The Binary Genetic Algorithm. In Practical Genetic Algorithms," *John Wiley and Sons Inc,* pp. 27-50, 2004.